P.O. BOX 1450
ALEXANDRIA, VA 22313-1450
IF UNDELIVERABLE RETURN IN TEN DAYS

OFFICIAL BUSINESS

02 1A $ 01.29
0004205065 OCT 04 2005
MAILED FROM ZIPCODE 22314

AN EQUAL OPPORTUNITY EMPLOYER

RECEIVED
OCT 11 2005
USPTO MAIL CENTER

AUST987 840101016 1804 03 10/07/05
FORWARD TIME EXP RTN TO SEND
AUSTIN
1244 E 1650 S
BOUNTIFUL UT 84010-1585

RETURN TO SENDER

2134
IFW

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/027,714	12/21/2001	David M. Austin	AUZ-002 P	6090

7590 10/04/2005

Wesley L. Austin, Esq.
1987 South Bluebell Drive
Bountiful, UT 84010

EXAMINER	
SZYMANSKI, THOMAS M	

ART UNIT	PAPER NUMBER
2134	

DATE MAILED: 10/04/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/027,714	AUSTIN ET AL.	
	Examiner	Art Unit	
	Thomas Szymanski	2134	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 December 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-34* is/are pending in the application.

4a) Of the above claim(s) *22-34* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-21* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) *22-34* are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *21 December 2001* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *12/21/2001*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Election/Restrictions

1. Restriction to one of the following inventions is required under 35 U.S.C. 121:

   I. Claims 1-21, drawn to a computer program for the detection of an observing program, classified in class 726, subclass 24.

   II. Claims 22-29, drawn to Generating system input and then monitoring associated activity for observer programs, classified in class 713, subclass 187.

   III. Claims 30-32, drawn to a ciphering program for ciphering the users keystroke input, classified in class 713, subclass 189.

   IV. Claims 33-34, drawn to a program for detection of network sniffers, classified in class 713, subclass 153.

The inventions are distinct, each from the other because of the following reasons:

2. Inventions I, II, III and IV are related as subcombinations disclosed as usable together in a single combination. The subcombinations are distinct from each other if they are shown to be separately usable. In the instant case, invention I has separate utility such as specifically scanning separate portions on the computer system for detection of observer programs, whereas invention II generates false activity and monitors associated responses. See MPEP § 806.05(d).

Because these inventions are distinct for the reasons given above and have acquired a separate status in the art as shown by their different classification, restriction for examination purposes as indicated is proper.

3. During a telephone conversation with Wesley Austin on 9/19/2005 a provisional election was made without traverse to prosecute the invention of I, claims 1-22. Affirmation of this election must be made by applicant in replying to this Office action. Claims 22-34 withdrawn from further consideration by the examiner, 37 CFR 1.142(b), as being drawn to a non-elected invention.

4. Claims 1-21 have been examined.

Specification

5. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

6. The applicant is requested to review the specification and update the status of all co-pending applications made mention of, replacing attorney docket numbers with current U.S. application or patent numbers when appropriate. References to U.S. applications or patents should make it clear as to what the number refers (e.g. U.S. Patent No. #), instead of listing only the number.

Drawings

7. Figures 1-2 should be designated by a legend such as --Prior Art-- because only that which is old is illustrated. See MPEP § 608.02(g). Corrected drawings in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. The replacement sheet(s) should be labeled

"Replacement Sheet" in the page header (as per 37 CFR 1.84(c)) so as not to obstruct any portion of the drawing figures. If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Rejections - 35 USC § 101

8. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

9. Claims 1-21 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. As stated the subject matter of the above noted claims refers to a computer program that is not stated as being contained within any tangible medium. In order for such subject matter to conform to the statutory basis it must be contained within a computer readable medium or some other form that is tangible.

Claim Rejections - 35 USC § 103

10. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

11. Claims 1-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Togawa U.S. Patent No. 6,240,530, and further in view of Drake U.S. Patent No. 6,006,328.

12. Togawa teaches a system for the detection and removal of computer malware.

13. Togawa fails to teach explicitly searching for observer programs as part of that malware.

14. Drake teaches security methods to protect against attacks on a computer system and its software from such sources as eavesdropping on those computer systems.

15. It is desirable within any computer system to maintain the security and integrity of such a system while preventing damage to the data and components included therein. (Drake Col 3 lines 30-52)

16. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the system of Drake with that of Togawa for the advantages of improved security by adding the features of protection against such malicious activities as eavesdropping to the ability of the scanning system as described by Togawa.

17. Regarding Claims 1 and 21: Observer program data characteristics (Togawa Fig 1.s1, Col 5 lines 10-19 Drake Fig 4,5 Col 3 lines 31-52) As it is understood the detection of a virus and its type as within Togawa requires recognition of characteristics of a virus. Those characteristics residing within the computer systems various components as any particular various infects that system; so then the same is true

within the combined system for the detection of an observer program as defined by Drake.

Obtain memory data of the computer (Togawa Fig 1, Col 8 lines 14-30) As explained above the detection of the malware requires checking the system which is inclusive of the memory data; therefore in order for the functionality to proceed it must in some way obtain such data for scanning.

Comparing memory data with observer program data characteristics for detection of an observer program (Col 8 lines 14-30) As it is known within the art virus scanning is the process of comparing two such sets of data. Further within the combined system the observer program characteristics are included within the set of the compared traits.

Generating a result of whether an observer program is present (Fig 1, Fig 3-4 Col 5 lines 10-38) Detection denotes that a result is generated as to the response of the scanning process.

Presenting results through a GUI (Fig 3-4, Col 5 lines 39-50, Col 13 lines 8-55, Col 14 lines 18-25) As denoted the display performs functions of disseminating operational information which is in a graphical form and presented within an OS that the user is capable of interacting with.

18. Regarding Claims 2 and 3: Memory data includes startup and registry startup commands (Col 8 lines 14-30, Col 13 lines 19-56) As stated the memory contains all necessary information for the processes of the machine; these processes being inclusive of starting up necessary portions for operation thereof; such as the OS which

includes a registry and the virus detection that being its own implementation scans the memory that these commands are located within.

19. Regarding Claims 4 and 5: Observer program characteristics include observer import/export table data for comparison with memory import/export table data to determine the presence of an observer program (Col 8 lines 14-30, Col 13 lines 19-56) As explained above all of the common features of the memory and functionality of the system are scanned via the anti-malware system.

20. Regarding Claim 6: Observer program characteristics include observer resource data for comparison with memory resource data to determine the presence of an observer program (Col 8 lines 14-30, Col 13 lines 19-56)

21. Regarding Claim 7: Observer program characteristics include observer file content data for comparison with memory file content data to determine the presence of an observer program (Col 8 lines 14-30, Col 13 lines 19-56) Additionally, as is shown and well known within the art file content is compared to malware characteristics for detection of such programs located commonly in such a place.

22. Regarding Claim 8: The comparing instruction compare the observer file content data with memory file content data at an offset address (Fig 1, Fig 3-4, Col 5 lines 10-20, Col 13 lines 19-56) The process of scanning for malware is inclusive of the entire range of memory; therefore the process must offset the data being scanned by that which has already been.

23. Regarding Claim 9: The comparing instruction compare the observer file content data with a span of the memory file content data identified by an offset address (Fig 1,

Fig 3-4, Col 5 lines 10-20, Col 13 lines 19-56) The process of scanning for malware is inclusive of the entire range of memory; therefore that which is scanned is a span of memory that is offset by the amount previously scanned.

24. Regarding Claim 10: Observer program characteristics include observer module loading data for comparison with memory module loading data to determine the presence of an observer program (Col 5 lines 10-20, Col 13 lines 19-56)

25. Regarding Claim 11: Observer program characteristics include OS observing functions for comparison with memory functions from the memory data to determine the presence of an observer program (Col 5 lines 10-20, Col 13 lines 19-56)

26. Regarding Claim 12: Memory data includes explorer extension data (Col 13 lines 19-56)

27. Regarding Claim 13: Memory data includes file use information (Col 13 lines 19-56)

28. Regarding Claim 14: Memory data includes process information (Col 13 lines 19-56)

29. Regarding Claim 15: Memory data includes running process information (Col 13 lines 19-56)

30. Regarding Claim 16: Memory data includes loaded module information (Col 13 lines 19-56)

31. Regarding Claim 17: Memory data includes driver data (Col 13 lines 19-56)

32. Regarding Claim 18: Memory data includes kernel driver data (Col 13 lines 19-56) All of the above stated separate memory data components are included within any

resident memory of a common computer system that a system such as the combination of Togawa and Drake would be implemented upon.

33. Regarding Claims 19 and 20: Instruction to disable an observer program if present (Fig 1, Fig 10, Col 5 lines 10-50, Col 19 line 15 – Col 20 line 65)

Entering a startup command to load a kill program before the observer program is started (Fig 10, Col 19 line 15 – Col 20 line 65) As shown within the figure the system clears the memory then loads a secondary extermination routine, inclusive of the secondary OS and associated extermination routine, so that the observer program is not reloaded and instead the kill program is loaded and executed.

Rebooting the computer (Fig 1, Fig 10) As it is shown after the detection and initial clearing of memory the system must be rebooted with a separate non-infected operating system to further allow for the deletion of any other virus elements.

Starting the kill program by execution of the startup command (Fig 10, Col 19 line 15 – Col 20 line 65) As explained above the kill program is loaded at startup so the virus may not load.

Deleting the observer program startup command and files (Fig 10, Col 19 line 15 – Col 20 line 65) The process of clearing the memory as stated within the cited lines and exterminating the malware is the process of deleting the startup command.

Conclusion

34. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection

of claims, the patentable novelty must be clearly shown in view of the state of art disclosed by the references cited and the objections made. Applicant must show how the amendments avoid such references and objections. See 37 CFR 1.111(c).

35. Inquiries concerning this communication or earlier communications from the examiner should be directed to Thomas M. Szymanski who can be reached at (571) 272-8574. The examiner's normal working schedule is between the hours 8:00am – 4:30pm (EST), Monday – Friday.

36. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Gregory Morse, can be reached at (571) 272-3838. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

37. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

JL

GREGORY MORSE
SUPERVISORY PATENT
TECHNOLOGY

#2

FORM PTO-1449 LIST OF PATENTS AND PUBLICATIONS FOR APPLICANT'S INFORMATION DISCLOSURE STATEMENT (use several sheets if necessary)	SERIAL NO.	ATTORNEY DOCKET NO. AUZ-002 P
	FILING DATE December 21, 2001	GROUP ART UNIT
	APPLICANT(S): David M. Austin et al.	

REFERENCE DESIGNATION U.S. PATENT DOCUMENTS

EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS/ SUBCLASS	FILING DATE
TMC	A1	6,006,329	12/21/1999	Chi	713/200	08/11/1997
	A2	5,978,917	11/02/1999	Chi	713/201	08/14/1997
	A3	5,964,889	10/12/1999	Nachenberg	714/25	04/16/1997
	A4	5,907,834	05/25/1999	Kephart et al.	706/20	03/18/1996
	A5	5,832,513	11/3/1998	Kennedy	707/202	06/04/1996
	A6	5,696,822	12/09/1997	Nachenberg	380/4	09/28/1995

EXAMINER	DATE CONSIDERED
Thuan Du	9/26/05

EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant(s).

Notice of References Cited	Application/Control No. 10/027,714	Applicant(s)/Patent Under Reexamination AUSTIN ET AL.	
	Examiner Thomas Szymanski	Art Unit 2134	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
	A	US-6,006,328	12-1999	Drake, Christopher Nathan	726/23
	B	US-6,240,530	05-2001	Togawa, Yoshifusa	714/38
	C	US-6,289,448	09-2001	Marsland, Tim P.	713/2
	D	US-6,021,510	02-2000	Nachenberg, Carey	714/38
	E	US-			
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N	EP 0449242A2	02-1991	EP	Watson et al	G06F 11/00
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 449 242 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **91104834.6**

(51) Int. Cl.5: **G06F 11/00**

(22) Date of filing: **27.03.91**

(30) Priority: **28.03.90 US 500755**

(43) Date of publication of application:
02.10.91 Bulletin 91/40

(84) Designated Contracting States:
AT BE CH DE DK ES FR GB GR IT LI LU NL SE

(71) Applicant: **NATIONAL SEMICONDUCTOR CORPORATION**
2900 Semiconductor Drive P.O. Box 58090
Santa Clara California 95051-8090(US)

(72) Inventor: **Watson, Bruce William**
5615 Fort Fisher Way
Norcross, Georgia 30092(US)
Inventor: **Lee, R. Jeff**
2581 Deer Isle Cove
Lawrenceville, Georgia 30244(US)

(74) Representative: **Sparing Röhl Henseler**
Patentanwälte European Patent Attorneys
Rethelstrasse 123
W-4000 Düsseldorf 1(DE)

(54) Method and structure for providing computer security and virus prevention.

(57) Tests are performed prior to or during the boot operation to determine whether files are corrupted. This may indicate the presence of a virus. If a potential error is detected, boot is halted, allowing the user to boot from uncorrupted files. In another embodiment, an uniquely formatted floppy diskette is used as an access diskette serves as a hardware key to gain access. In another embodiment, a host controls information stored locally. In another embodiment, security from unauthorized access is provided once a valid user has legitimately accessed a computer. In response to a predefined hot key or a predetermined period of time during which the user has not provided input, portions of the computer are disabled. Upon entry of access information by the valid user, the disabled features are enabled. In another embodiment, access to the computer is made more difficult in response to invalid access attempts. In one embodiment, once a threshold number of invalid access attempts is reached, the computer is locked up, requiring reboot, thereby increasing the difficulty of a would be intruder to gain access to the computer. In one embodiment, once the threshold value is reached, it is reset to a lower value.

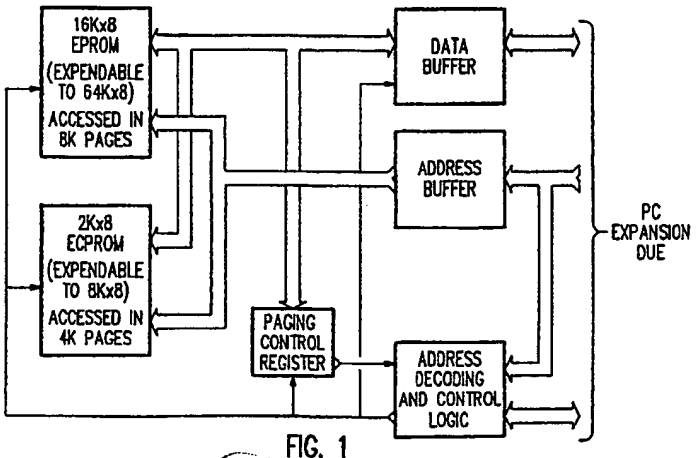

FIG. 1

EP 0 449 242 A2

Although password access control exists on a host system, which is administered by the host, no mechanism currently exists for a host system to remotely control access to a personal computer.

There are a number of other prior art approaches to PC security. They span the gamut from biometric thumb scan access, to special sealed Tempest shielded PCs to prevent electronic snooping, to DES encryption of all files with specific access passwords on a file or directory level. While such severe measures are certainly warranted in the use and handling of highly sensitive or classified data, this level of security is not generally required or needed in the business environment.

Another threat to the data and programs stored on a PC is attack from a computer virus. These small programs can attach themselves to a normal program or data file, which may be inadvertently copied from system to system. Once a virus program has been copied to a PC, it can cause serious damage to program and data files that reside on the system. Without some form of protection, the PC user may not realize that the system has been infected until after damage has been done.

There are a number of prior art approaches to virus protection. They include never using a disk from anyone, never downloading data from a bulletin board service (BBS) or via a modem, not connecting to a network, checking each program or file with a virus finder, only running new programs on a floppy or on a quarantine machine until determined safe, having a special program slow down the processing of the PC while it watches every activity to determine if is normal or a virus. While these measures can add additional levels of protection, practicality in a business environment must be considered.

Some prior art approaches to virus protection check hard disk system ares and files after the disk operating system has booted. Any damage caused by corrupted system files and data may already be done. If a virus has already attached itself to one of the operating system files, once this file has been loaded (during boot), the virus is in control.

Other approaches to virus protection require booting the system from a known floppy disk and checking the hard disk before allowing the use of files stored on the hard disk.

As other approaches to virus protection are resident on disk, they themselves can be corrupted.

Even if you have protected your system from unauthorized access and viruses, once your PC is turned on and validly accessed, it is again vulnerable. Most people turn their PCs on in the morning and leave them on all day. Lunch, an out of the office appointment, long meetings, or just going down the hall are all opportunities for unauthorized access to data. A quick erase or a quick copy to a diskette may leave no signs of activity. However, it is impractical to exit your program, save your data, and turn your PC off every time you leave your desk for a few moments.

A PC screen can also inadvertently expose sensitive data such as payroll, financial, or personnel records to anyone. Information left on the screen while at lunch or a meeting, or obtainable through just a few keystrokes, is all that may be required to steal or destroy a file. In today's aggressive business environment, a company's competitive edge is relative to its customer files and information, or proprietary information such as product development, marketing and sales plans, and product price and costs lists.

Accordingly, it remains highly desirable to provide the ability to detect the presence of viruses in a manner which will allow the operation of the computer to be halted, for example by failing to complete the boot operation, on a regular basis. Furthermore, computer security remains of vital importance.

SUMMARY OF THE INVENTION

In accordance with the teachings of this invention, various techniques are employed in order to provide security to a computer system. Certain embodiments are particularly well suited to network environments, and in particular to network environments including one or more PCs. In one embodiment of this invention, tests are performed prior to or during the boot operation in order to determine whether selected programs and/or data files have been corrupted. This may indicate the presence of a virus. In one embodiment, files which will be used during the boot operation are checked for modification prior to allowing those files to be used for system boot. If a potential error is detected, system boot is halted, allowing the user to boot the system from a known, uncorrupted set of files, for example as contained on a floppy diskette. In an alternative embodiment of this invention, additional program and/or data files (which are not used during the boot operation) are checked for corruption either prior to the boot operation or immediately following the boot operation.

In another embodiment of this invention, a unique access diskette is used as a hardware key to allow a user to demonstrate his authorization to gain access to the computer. The unique hardware key is provided by uniquely formatting a standard floppy diskette in such a manner that it contains information indicating the user's authorization to access the system. In one embodiment, this unique formatting is such that the information contained on the access diskette cannot be easily read using the standard diskette reading

intelligent log on to a computer system.

While certain specific embodiments described herein refer to providing security to PC systems, and PC systems connected in a network, perhaps with a host computer such as a mainframe, it is to be understood that the teachings of this invention are equally applicable to a wide range of computer applications, including computer networks which do not utilize PCs.

Furthermore, a number of specific embodiments are described below. It will be appreciated by those of ordinary skill in the art in light of the teachings of this invention that various combinations of these embodiments may be used in any particular system, or may be provided in a system for selection by a user, thereby providing a large number of permutations of combinations of the various features of this invention.

I. Firmware Virus Checker

Figure 1 depicts one embodiment of a security hardware subsystem of this invention which implements an 8KB aged ROM window. This embodiment allows a 16KB EPROM (which contains the virus checker algorithm and, if desired, other various security related firmware) and a 2KB EEPROM (a non-volatile memory device which is used to store security configuration data) to be accessed by the computer from within a single 8KB window. This embodiment minimizes the requirements for system memory space, while also providing an indirect (hidden) access mechanism for the EEPROM device (i.e. protecting secure data).

The following description refers to a computer system including a mass storage device, such as a hard disk as found in a typical PC. It is to be understood that the teachings of this invention apply equally well to systems including one or more hard disks, virtual disks, or hard disks partitioned as more than one disk. Furthermore, the teachings of this invention apply equally well to systems utilizing other types of storage media.

This invention is particularly well suited for use with desktop or laptop computers which are perhaps more susceptible to viruses than large systems which incorporate sophisticated security schemes. In accordance with the teachings of this invention, a firmware resident program is provided which accesses the file structure of the hard disk before the disk operating system is loaded, in order to verify the integrity of the data and program files on that disk which will be used during the disk operating system boot loading process. In one embodiment of this invention, the system areas checked in a typical IBM compatible PC is shown in Table 1.

TABLE 1

System CMOS SRAM
Disk Boot Areas (Hard Disk Master Boot Track and the DOS Boot Sector)
DOS BIO hidden system file (IO SYS or IBMBIO.COM)
DOS OS hidden system file (MSDOS.SYS or IBMDOS.SYS)
DOS command processor file (COMMAND.COM)
AUTOEXEC.BAT
CONFIG.SYS

In an alternative embodiment, the method of this invention verifies the integrity of all or a selected set of program and data files in addition to those programs and data files which are used during the disk operating system boot loading process. Since the program of this invention is contained within the computer itself, it eliminates the need to boot from a floppy disk in order to pre-check the hard disk. Furthermore, in one embodiment, the virus checking software is resident in firmware, such as a ROM, and thus occupies no disk space and eliminates the possibility of the virus checking software of this invention itself being corrupted by viruses.

compared with the old CRC and differences are posted.

10. The CONFIG file is read into a contiguous buffer and a CRC is calculated. The new CRC is compared with the old CRC and differences are posted.
11. If any errors are detected, system boot is halted in order to allow the user the opportunity to boot from a recovery diskette or alternate DOS diskette, if desired.

Once the algorithm of this invention has been executed, control is returned to the system BIOS to allow continuation to the normal DOS boot process, as shown in Fig. 2.

When the program of this invention is executed prior to disk operating system boot, each of the system data areas and files are scanned, and signatures calculated. These new signatures are then compared with the values stored in non-volatile memory. Any differences are reported as an integrity failure with specific data or program files. If a failure is detected, the user is given the option to boot from a recovery diskette, thereby allowing the system to be booted from a disk containing a disk operating system known to be uninfected by a virus. Thus, the computer becomes usable and the user is able to investigate the potential problems on the hard disk, and allows the replacement of potentially infected or corrupted program and data files. By restoring corrupted files prior to their use during boot or subsequent execution of applications software, possible system disaster can be averted by preventing the spread of a virus throughout the system.

This teachings of this invention are suitable for use in any type of device which loads critical operating system information from hard or soft disk media, where potential contamination by a virus could have dangerous impact, for example: process control, power plants, launch control, lottery, communications center, data routing systems, electronic funds transfer, and the like.

In one embodiment, specific program or data files which have been previously specified by a user are automatically checked after the system has booted. If any of the selected files have been modified, indicating a potential problem, including the presence of a virus, the user will be notified of the change.

II. Access Diskette

In accordance with the teachings of this invention, a diskette is uniquely formatted using a method which is non-standard for the resident disk operating system. This diskette is then capable of being read and verified by resident security software so that it may be used as a hardware key to access the system. Of importance, in accordance with teachings of this invention, a diskette key is provided which utilizes standard floppy disks, rather than special hardware or key devices as known in the prior art. However, the standard floppy disk which is formatted according to the teachings of this invention in order to serve as an access key cannot be read or copied by standard operating systems utilities or after-market disk utility programs.

In one embodiment of this invention, the algorithm used to create access diskettes is based on two randomly generated signatures (numeric values) such that all access diskettes (with the exception of a backup access diskette, if desired) are unique in two ways:

1. Each access diskette is uniquely formatted; and
2. Each access diskette contains different "key" information.

In one embodiment, the signature values which are used to create access diskettes are generated such that they are unique for a given PC, and unique from one creation time to another. If desired, a mechanism is also provided to manually supply the signature values during access diskette creation.

The access diskette is read and verified by resident software (or firmware) which requires a valid access diskette before access to the system is allowed. The access diskette may be used as a "stand-alone" key to the system or may be used in conjunction with a password to "double-lock" the system.

In accordance with one embodiment of this invention, a password is used in conjunction with the access diskette in order to control access to the computer. In an alternative embodiment of this invention, certain passwords (e.g. a User password) are sufficient to obtain access the computer without the use of the hardware access diskette. The access diskette must be used, however, to obtain a higher level of access, for example that level which is available when utilizing a Supervisor password.

An access diskette of one embodiment of this invention comprises three parts:

(1) Identification information indicating this is an access diskette. This identification information is stored in a portion of the diskette which will be referred to as the identification sector or identification track.

(2) An access signature, which is preferably unique for a given system. The access signature is stored in a portion of the access diskette called the key sector or key track.

(3) The remainder of the diskette.

The identification information indicating this is an access diskette need not be used for the purpose of

access diskette is created.

In one embodiment, the access signature is created as the current (at the time the access diskette is created) CRC of a portion of the system data. For example, a CRC of the file allocation table (FAT) may be created, or of a specified portion of the hard disk, or the RAM, or a specified portion thereof. Alternatively, additional hardware memory (such as the non-volatile memory used for storing security information, such as is described above with regard to virus checking) is used to create the access signature. Using the current CRC as a location code insures that the creation of each key disk is unique. A lookup table is provided such that each authorized key disk value is stored and checked upon insertion of a key disk for login purposes. A system administrator is able to delete undesired previous key disk access signatures as they become no longer required.

In one embodiment, the remainder of the access diskette, i.e., that portion of the access diskette which is not used to store the identification information notice or the access signature, is ignored, leaving a large number of unused tracks. In an alternative embodiment, the remainder of the access diskette is filled. In this embodiment, the unused tracks are preferably filled such that the diskette contains a large number of unused tracks which are filled with data which appear substantially similar to the data stored in the tracks used to store the identification information notice and access signature. For example, each unused sector is written to store a pattern similar to that stored in the key sector which stores the access signature except, of course, that the access signature is not contained within the key sector. This makes it very difficult for an unauthorized person to read the access diskette and determine the access signature.

III. Local Information Modified or Controlled by Remote Host

In accordance with one embodiment of this invention, a unique process is provided by which a remote host computer system may access a local desktop or personal computer and modify its information. This information may be data, whereby the host computer can modify local computer data, for example as the master information stored in the host changes, or in order to encrypt or decrypt information which is stored in the local computer. Alternatively, or in addition thereto, this information which is capable of being modified or controlled by the remote host computer can be local security information, thereby allowing the host to remotely control access to the PC. This allows system administrators to control access to individual PCs remotely, via the host computer system. The teachings of this invention may be used to remotely change information stored in the PC which determines whether a user may gain access to that PC, such as authorized user ID and password information. It is also possible for the host to update information stored within the PC which serves as a table of information which is authorized to be accessed by one or more users. System administrators may also use the remote system of this invention to remotely access use information stored within the PC, such as date and time information associated with each successful and/or unsuccessful attempt to log into the PC.

Figure 4 is a flow chart depicting one embodiment of this invention for use in a network including a host computer and a PC having local security information which is capable of being modified from the remote host. As shown in the flow chart of Figure 4, a first step in this embodiment is that the PC logs into the host system, for example in any one of a number of well known ways, following which the PC is connected to the host computer as shown in Step 2. In Step 3, the host modifies or updates selected information in the PC, such as the user ID and/or password information of valid users. In Step 4, the host checks the audit log of the individual user of the PC system in order to determine user audit log information. In the event that the audit log indicates unauthorized activities, the host computer causes user access to be revoked.

Conversely, if the check of the audit log of the individual PC user determines no reason to forbid access to the host computer, operation branches to box 5 in which the host computer requests and receives from the PC a password from the PC user. In Step 6, the host computer modifies a list of allowable data files which may be accessed by one or more users. Secure data associated with the user, as shown in Step 7, is downloaded from the host to the PC. This secure information is appropriately stored within the PC for later use by the PC.

In one embodiment, as shown in Step 8, this secure data is not stored on the disk of the PC where it might be undesirably retrieved by a potential intruder. This allows greater control of secure information since the data stored in memory is erased when the PC is turned off. In this embodiment (for example as shown in Fig. 1), such security information is stored in hardware contained within the PC, for example in nonvolatile memory (such as an EEPROM) located in an area of the PC dedicated to security functions, such as a security card installed in one of the expansion slots.

As shown in Step 9, the host computer then allows the PC to be used as a work station with access to the host computer.

login audit log information is, in one embodiment, also available for remote polling by a host system, thereby allowing a system administrator to become alerted to attempted breaches of security at the PC level.

Conversely, if in Step 3 an invalid login attempt is detected, control passes to decision Step 5 which determines if this is the first invalid login attempt during a specified period of time, for example since the last valid login, or during a particular time period. If it is determined that this is the fist invalid login attempt according to the criteria used by decision Step 5, control passes to Step 6 and the computer beeps twice as an indication of an invalid login attempt. This audible signal is not too distracting but may alert others in the area to pay particular attention if additional error signals are heard. Control is then passed to block 2, allowing the user to attempt to login again. Conversely, if Step 5 determines that an invalid login attempt was not the first, control is passed to decision Step 7 where it is determined if the invalid login attempt was the second invalid login attempt. If so, control passes to Step 8, wherein the computer beeps several times prior to passing control back to Block 2. This will allow others in the vicinity to notice that something is awry and perhaps call security or take their own steps in order to determine the identity of this would-be intruder.

Conversely, if decision Step 7 determines that the invalid login attempt was not the second invalid login attempt, control is passed to block 9 which initiates action commensurate with the fact that three or more invalid login attempts have been detected. Control is passed to block 10 wherein the computer system is locked, for example by halting the system microprocessor, requiring the system to be rebooted (either hot or cold, as specified by information to which block 10 responds which information has been previously supplied by a legitimate user). This requires that the computer be rebooted, as shown in Step 11, prior to returning control to Step 2, which again allows a user to attempt to obtain access. Upon reboot, the threshold is set to one, thereby allowing only one password access failure before locking up the system and thereby requiring a reboot. Thus, after the initial threshold value (in this example, 3) of unsuccessful login attempts is reached, the system is locked up and must be rebooted, after which only one invalid login attempt is allowed before requiring the system to be rebooted. In one embodiment of this invention, the programmable threshold value is reset to its normal value (in this embodiment, 3) in response to to a valid login.

By requiring the system to be rebooted after a predetermined number of failed login attempts, considerable difficulty is encountered by the unauthorized user who is attempting to illegally gain access to the system. For example, rather than being able to rapidly enter a large number of potentially correct passwords on a hit or miss basis, the process is considerably slowed by the need to reboot the computer after three failed login attempts. In addition, audible signals from the computer indicating failed attempts will alert others in the physical proximity of the computer that something inappropriate is happening.

Naturally, other embodiments of this invention will become readily apparent to those of ordinary skill in the art in light of the teachings of this invention, for example, additional decision steps such as steps 5 and 7 may be employed so that any number of security threshold levels may be provided. For example, in one alternative embodiment, no audible signal is provided in response to a first invalid login attempt, as often occurs due to a simple typographical error by a user during his login attempt. In response to a second invalid attempt, the computer beeps twice; in response to third, fourth, and fifth invalid attempts, the computer beeps five times; and in response to the sixth invalid attempt the computer locks up, requiring reboot. This as well as numerous other variations are all within the scope of this invention.

Other alternative embodiments include not merely increasing the number of audible tones in response to an increasing number of failed logon attempts, but varying their intensity, pitch, or the like. For example, in response to an additional one or two failed logon attempts, a simple beep may be sounded while in response to a third failed logon attempt, a lengthy and loud wailing sound might be produced. Furthermore, in addition to audible signals, unique visual patterns may be flashed upon the screen, perhaps frightening the intruder away, and possibly alerting others in the area that something inappropriate is occurring on the computer.

VI . Non-Volatile Security Configuration Data Storage

A mechanism is provided for the storage of security configuration and audit data in a non-volatile storage device, such that the data can be error checked and corrected and protected from tampering.

In this embodiment, security data is stored in non-volatile memory (e.g. hardware incorporated in the security hardware shown in Figure 1) which eliminates the need to store on an easily accessible hard disk.

In this embodiment, data is structured in redundant tables whose contents are monitored via both checksum and CRC algorithms. By containing embedded checksums and CRC, the data can be readily checked for corruption. The redundant table structure provides a mechanism for automatic data recovery

allowing access if said access diskette contains valid authorization information.

11. A method for storing security information in a computer network comprising a plurality of computers, comprising the steps of:
causing a first one of said computers to transmit data for storage in a second of said computers, said information serving to control user access to said second computer.

12. A method for controlling a computer comprising the steps of:
during operation of the computer, detecting the occurrance of either:
entry of a selected command from a user; or
an elapse of a selected amount of time during which there has been no data or commands inputted by the user;
in response to said occurance, performing the following steps:
disabling one or more input/output functions of said computer;
monitoring for user input of information indicating valid user authorization for use of said computer; and
in response to the detection of valid user authorization, enabling said one or more input/output functions of said computer which had previously been disabled.

13. A method as in claim 12 wherein said one or more input/output functions are selected from the group of functions consisting of keyboard entry, data entry, screen display, printer output, and data transmission.

14. A method as in claim 12 wherein selected functions of said computer are not disabled in response to said occurance.

15. A method as in claim 14 wherein said selected functions comprise computer processing.

16. A method as in claim 14 wherein said selected functions comprise execution of programs currently being run, or in a queue to run, at the time said occurance is detected.

17. A method as in claim 12 wherein said information indicating valid user authorization comprises a hardware access key, user identification, password information, or a combination thereof.

18. A method for controlling access to a computer comprising the steps of:
receiving user authorization information from a prospective user;
determining if said user authorization information is valid;
if said user authorization information is valid, allowing said prospective user to have access to said computer; and
if said user authorization information is invalid, increasing the difficulty of gaining access to said computer.

19. A method as in claim 18 wherein said step of increasing the difficulty of gaining access to said computer comprises the steps of increasing the amount or intensity of error signals provided in response to invalid user autorization information.

20. A method as in claim 18 wherein said step of increasing the difficulty of gaining access to said computer comprises the step of requiring said computer to be rebooted in response to a number of invalid user authorization attempts exceeding a threshold number of invalid user autorization attempts.

21. A method as in claim 20 wherein said threshold number is reduced to a lower threshold number following a number of invalid user authorization attempts which equals or exceeds said threshold number.

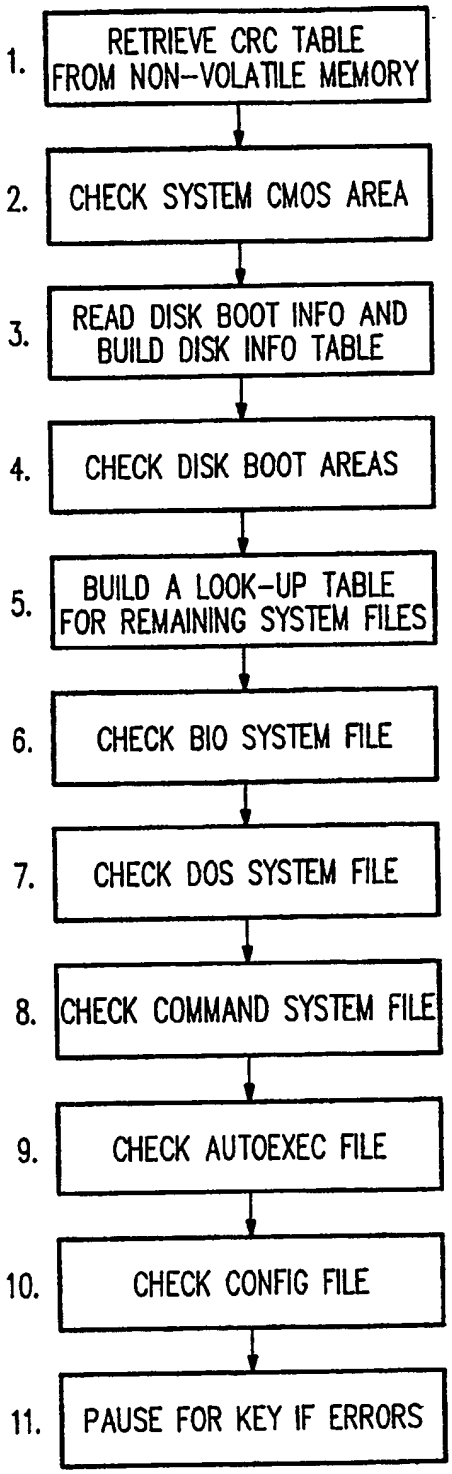
1.
RETRIEVE CRC TABLE FROM NON-VOLATILE MEMORY
2.
CHECK SYSTEM CMOS AREA
3.
READ DISK BOOT INFO AND BUILD DISK INFO TABLE
4.
CHECK DISK BOOT AREAS
5.
BUILD A LOOK-UP TABLE FOR REMAINING SYSTEM FILES
6.
CHECK BIO SYSTEM FILE
7.
CHECK DOS SYSTEM FILE
8.
CHECK COMMAND SYSTEM FILE
9.
CHECK AUTOEXEC FILE
10.
CHECK CONFIG FILE
11.
PAUSE FOR KEY IF ERRORS

FIG. 3

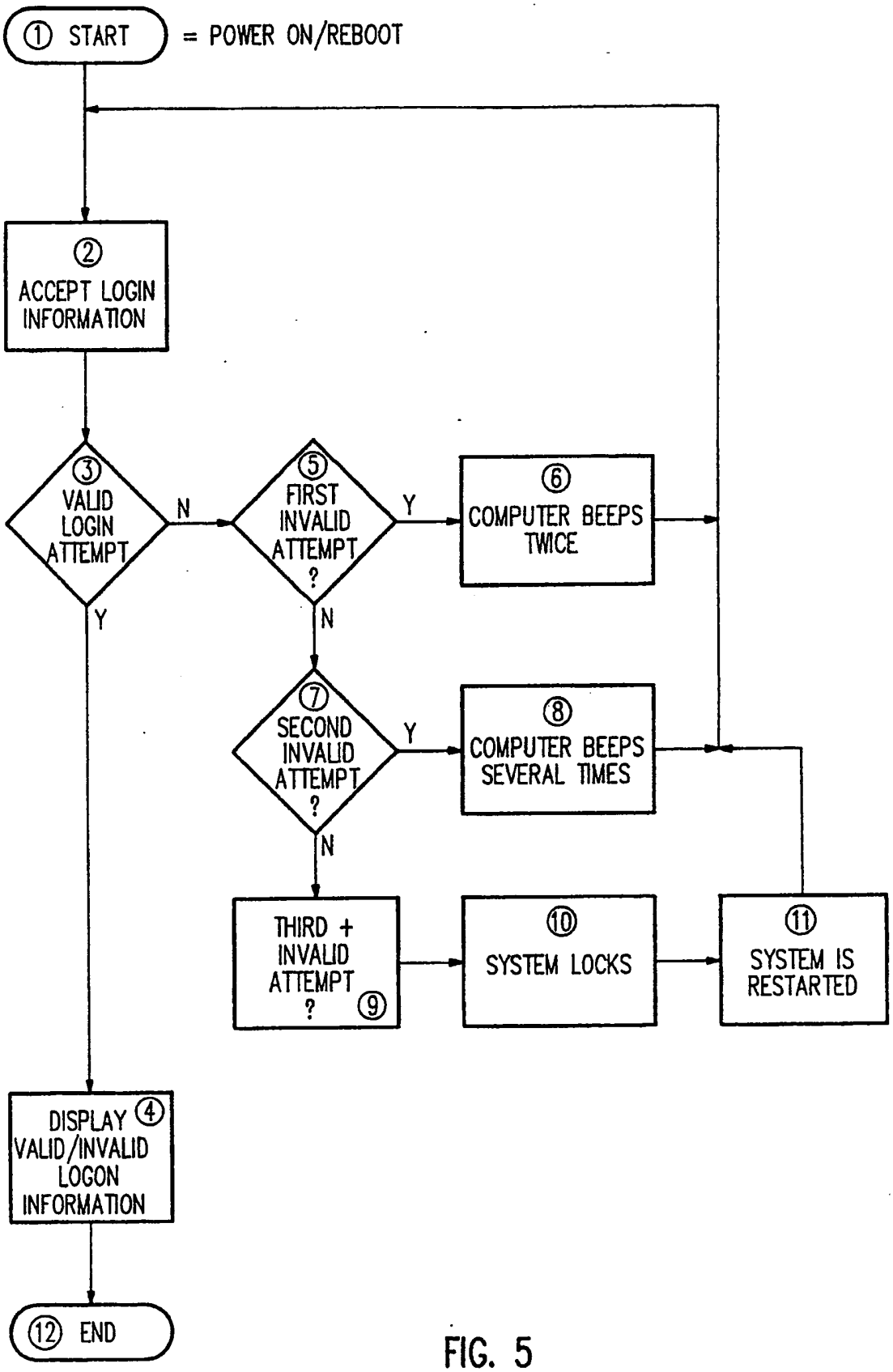
① START = POWER ON/REBOOT
② ACCEPT LOGIN INFORMATION
③ VALID LOGIN ATTEMPT
N
⑤ FIRST INVALID ATTEMPT ?
Y
⑥ COMPUTER BEEPS TWICE
Y
N
⑦ SECOND INVALID ATTEMPT ?
Y
⑧ COMPUTER BEEPS SEVERAL TIMES
N
THIRD + INVALID ATTEMPT ?
⑨
⑩ SYSTEM LOCKS
⑪ SYSTEM IS RESTARTED
④ DISPLAY VALID/INVALID LOGON INFORMATION
⑫ END

FIG. 5